16159 Postel

NIC 16159
RFC/NWG# 508

RECEIVED
JUN 6 1973

Computer Systems Laborator
UCSB
Larry Pfeifer
John McAfee
7 May 1973

REAL-TIME DATA TRANSMISSION ON THE ARPANET

I. INTRODUCTION

The ARPA Network is rapidly proving to be a useful tool in computer communications and resource sharing. It has been proposed that the same network might also be able to support real-time processes such as audio or video communications for conferencing purposes. The degree of support of these types of processes will largely be determined by transmission bit-rates and delays.

The IMP subnetwork throughput rates (one way) average about 37 kilobits[1], therefore an external process must operate at a bit-rate below that level. This would imply some form of data compression for both audio and video transmission. Research in these areas is still in progress so these processes must be simulated at the present time.

In addition to bit-rate, system response time (system delay) is an important factor since this has a direct influence on the amount of data which must be buffered in order to keep a real-time process running without discontinuities or gaps. Such delays may be caused by network loading, host loading, or an excessive number of IMP-to-IMP hops in the transmission path.

In order to get a feel for the ability of the network to support a real-time process an experiment was conducted with real-time data being sent from the UCSB SEL810-B computer, by way of the UCSB IBM 360 host, onto the ARPA Network and into a host discard socket in the UCLA IBM 360 computer. This particular

data path very nearly duplicates the path which might be taken if real-time devices were attached to large scale host computers operating in their normal mode (usually timesharing). The experiment consisted of measuring the duration of gaps incurred at various process bit-rates, and buffer sizes ranging from one to eight network packets.

Earlier experiments at MIT[2] simulated vocoded speech transmission over the ARPA Network using the TX-2 computer and "Fake host 3" in a destination IMP. Speech was sampled by the TX-2 and simulated speech data blocks were sent to a particular fake host. Receipt of an acknowledgment by TX-2 indicated that the corresponding blocks of speech data could be reconstituted. Experiments were conducted with bit-rates from 2400-17000 bps and varying block sizes (depending on the number of hops), and conclusions were reached that with delay characteristics similar to a lightly loaded ARPA Network speech communications could be satisfactory from a human-factors standpoint.

II. CONFIGURATION

Data for this experiment originated in an SEL 810-B computer located in the Electrical Engineering Department at UCSB. This 750ns cycle time computer is the heart of an interactive signal processing system developed by Retz[3]. It has associated hardware such as a card reader, two IBM 1311 disk drives, a drum storage unit, A/D and D/A converters, Teletype, Tektronix 611

storage display unit, OLS keyboard, and a connection to an IBM 1800 computer. This system is linked to the UCSB IBM 360/75 via a 500 kilobit line for high speed data transfers. Software in both the SEL 810-B and the IBM 360 enables the SEL to communicate with the ARPA Network.

The hardware configuration of the data path between the SEL 810-B and UCLA is shown in Figure 1. For simulating speech transmission, the SEL is thought of as a "speech processor", analyzing and encoding the one-way conversation of a person at UCSB talking to someone at UCLA. The fact that there was no "speech processor" at UCLA probably had little or no effect on the measurements that were made. This is substantiated by noting that the SEL was a dedicated processor that did not introduce delays and if a similar dedicated processor was attached to the host computer at UCLA it probably would not have caused delays either. However, the UCLA host merely discarded the data it received, thereby going through fewer steps than if an external processor was attached, and so our simulation was not exact.

A configuration such as that of Figure 1 did yield information about host-to-host transmission, since the SEL was essentially a zero-delay data generator. If real-time processors are to access the ARPA Network through large-scale time-shared host computers then host-to-host transmission rate and delay are important measurements. In this configuration we can expect the host computers to be the primary bottlenecks in the data path.

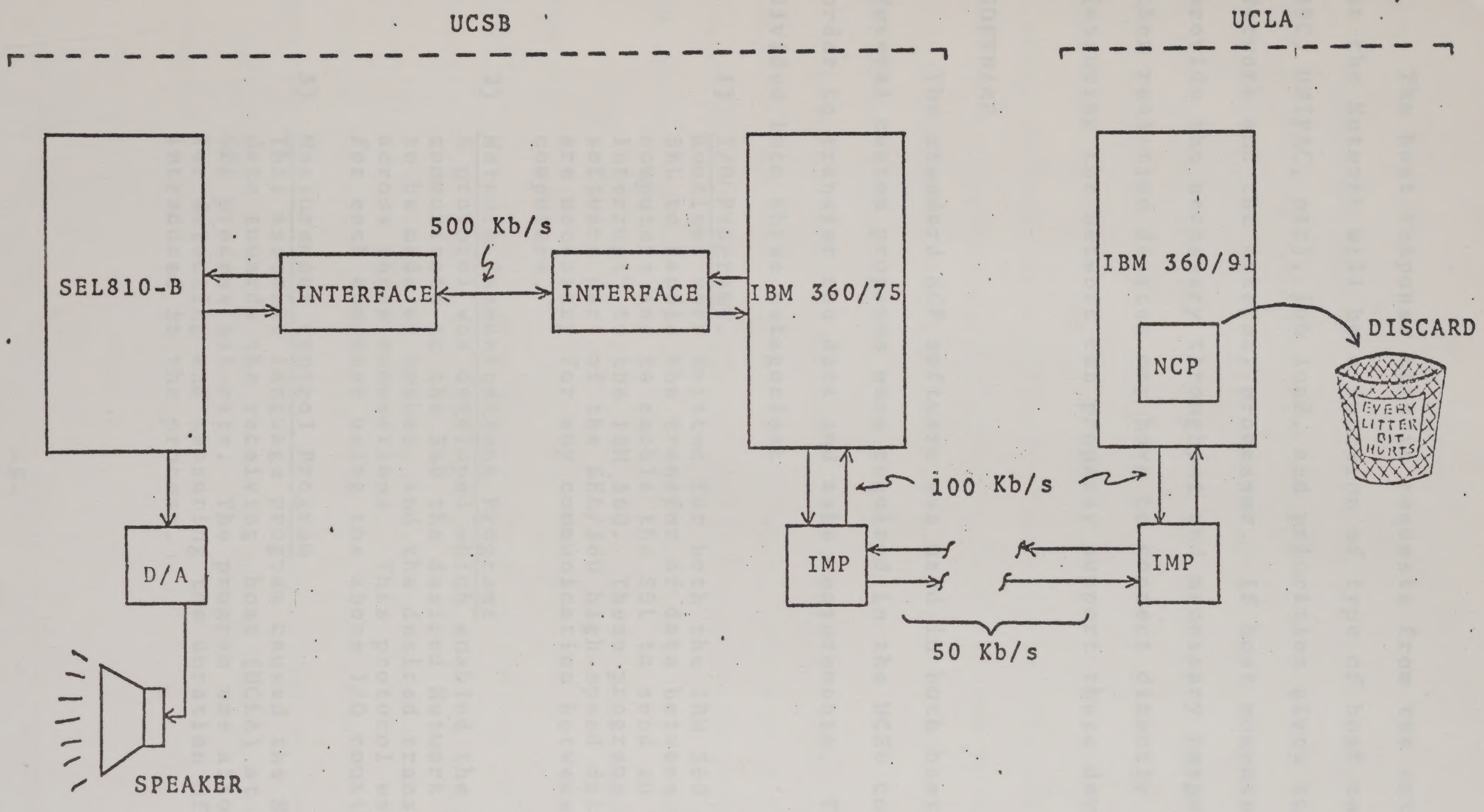

UCSB
UCLA
SEL810-B
INTERFACE
500 Kb/s
INTERFACE
IBM 360/75
IBM 360/91
NCP
DISCARD
EVERY LITTER BIT HURTS
100 Kb/s
IMP
IMP
50 Kb/s
D/A
SPEAKER

Figure 1. Hardware configuration of data path used for sending real-time data from the SEL 810-B to the UCLA host discard socket.

The host response time to requests from the external processor or the Network will be a function of type of host computer (IBM, DEC, UNIVAC, etc), job load, and priorities given to both the Network and the external processor. If host computers cannot provide the necessary throughput and necessary response times, then real-time devices may have to connect directly to IMPs (assuming the Network can properly support these devices).

III. SOFTWARE

The standard NCP software was used in both host computers. Several custom programs were required in the UCSB computers in order to transfer the data and make measurements. These can be divided into three catagories:

1) I/O Programs.
   Routines were written for both the IBM 360 and the SEL to handle the transfer of data between the two computers and to enable the SEL to send an "attention interrupt" to the IBM 360. These programs form the software part of the SEL/360 high-speed data link and are necessary for any communication between the two computers.

2) Network Communications Programs
   A protocol was developed which enabled the SEL to communicate to the 360 the desired Network connections to be made or broken, and the desired transfer of data across these connections. This protocol was implemented for each computer using the above I/O routines.

3) Measurement Control Program
   This assembly language program caused the SEL to push data towards the receiving host (UCLA) at a specified SEL process bit-rate. The program was also responsible for detecting and measuring the duration of any gaps introduced in the process.

IV. METHOD

Within the SEL two buffers, each of 1 to 8 network packets in length, are first loaded with alternating bit patterns in consecutive 16-bit words. A conversion process is then initiated on one of the buffers at a sampling frequency necessary to give the desired bit-rate. Since data is being sent out to a destination host we would expect the buffers to be filled by an analog-to-digital conversion process. However, in this experiment, the process of digital-to-analog conversion is used instead so that we can listen to the alternating bit patterns as a steady tone while still simulating an A-to-D process.

When a buffer is filled (played out) a "write" operation is initiated to send that buffer to UCLA. The next buffer is then tested to see if the previous "write" has been completed, i.e. the buffer is empty. If the next buffer is empty the process continues normally. If the next buffer is not empty it means that one of the computers on the Network has not taken the data fast enough, therefore a gap has been introduced in the real-time process. At this point the D-to-A converter is shut off resulting in an audible break in the tone that is being played out. A timer is also started to test for the empty buffer every one millisecond and to measure the duration of the gap. When the next buffer is finally emptied the D-to-A process is resumed and the gap data recorded in a table.

V. PROCEDURE

A connection to the UCLA host discard socket was first established using the network communications programs. Every test from this point on required a repetition of the following steps.

1) Initialize the UCSB IBM 360 for double buffered data transfers using specified buffer sizes.
2) Initialize the SEL measurement control program with the proper buffer size and process bit-rate.
3) Start the test. A constant tone from the speaker indicates that the process is being properly maintained. Gaps in the tone indicate gaps in the process.
4) After 30 seconds, stop the test.
5) Examine the gap table to determine the number of gaps, the duration of each gap, and the average duration.

The entire procedure was carried out from the SEL end using the interactive On-Line System. The timing interval of 30 seconds was measured with a sweep second hand of a watch and the test was started and stopped manually. All tests were conducted during prime time to obtain typical loading conditions.

VI. RESULTS

A total of 179 tests were conducted. Of these, 176 were 30 second tests and three were long duration tests. Table I contains the results of the 30 second tests. Buffer sizes were

varied from one to eight Network packets and for each buffer setting 22 different process bit-rates (usually in increments of 1200 bps) were attempted. These measurements were performed over a period of three days during prime time.

Those test conditions which were successful contain only two items of information in Table I: time of day and number of buffers transmitted. All but seven of the tests were successful. The tests which were unsuccessful, i.e. experienced gaps, are those entries in Table I which contain additional information such as number of gaps, and maximum, average and minimum gap duration.

An examination of those tests which failed shows that the longest gap which occured was 8 seconds in duration. There were three other significant failures between 9:52 A.M. and 9:59 A.M. on 2/7/73. There are strong indications that it was the UCSB 360 that caused these gaps to occur. This conclusion is based upon the fact that the Electrical Engineering On-Line classroom (16 interactive graphics terminals) was in full use that day until 10:00 A.M. and the SEL connection to the IBM 360 has lower priority in the 360 than the UCSB On-Line System. The remaining three tests which failed did not do so at any regular time, bit-rate, or buffer size so no definite statements can be made about their source of delay.

The overall picture presented by Table I is very promising. In 96 percent of the trials a communications link of two host computers and a portion of the ARPA Network was able to take data

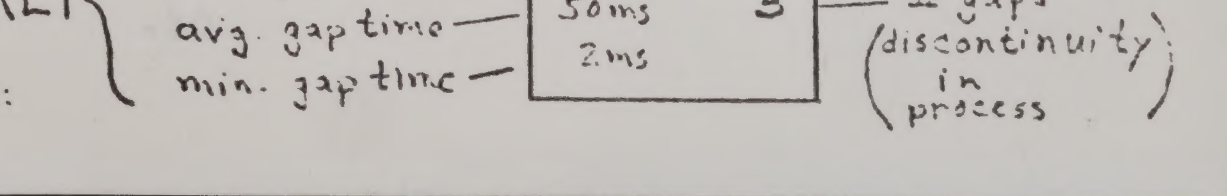
PROCESS BIT-RATE	BUFFER SIZE (BITS) 1008	2016	3024	4032	5040	6048	7056	8048
5300	11:31 158	9:51 81	11:34 45	10:34 41	10:12 33	1:59 28	1:37 24	12:32 21
6000	11:32 180	9:52 89 174 ms 1	11:40 61	10:35 46	10:13 37	2:00 31	1:38 27	12:36 23
7200	11:33 216	9:54 109	11:41 72	10:36 54	10:14 44	2:01 37	1:39 33	12:37 28
8400	11:34 245	9:55 126	11:42 82	10:37 63	10:14 51	2:02 42	1:40 37	12:38 32
9600	11:35 287	9:56 83 8 sec. 1	11:43 97	10:38 73	10:15 58	2:03 49	1:41 42	12:39 36
10800	11:36 318	9:57 138 3 sec 1.5 sec 2 100 msec	11:44 109	10:39 81	10:16 65	2:04 56	1:42 47	12:40 42
12000	11:37 358	9:58 180	11:45 119	10:44 91	10:17 73	2:05 61	1:43 52	12:41 46
13200	11:38 376	9:59 183 438 ms 269 ms 2 100 ms	11:46 132	10:45 101	10:18 80	2:06 67	1:44 57	12:49 49
14400	11:39 428	10:45 213	11:46 141	10:46 107	10:18 88	2:07 73	1:45 62	12:50 56
15600	11:39 467	10:46 232	11:47 156	10:47 117	10:19 94	2:08 79	1:46 67	12:51 59
16800	11:40 503	11:15 243 190 ms 128 ms 3 29 ms	11:48 168	10:48 127	10:20 100	2:09 85	1:47 72	12:52 63
18000	11:41 535	11:17 266	11:48 179	10:49 136	10:21 107	2:10 90	1:48 76	1:00 68
19200	11:42 573	11:18 285	11:49 191	10:50 144	10:22 114	2:11 98	1:49 82	1:20 73
20400	11:42 610	11:19 303	11:50 202	10:51 153	10:23 123	2:12 103	1:50 87	1:21 75
21600	11:43 643	11:20 327	11:51 213	10:52 162	10:24 130	2:13 108	1:51 94	1:22 81 98 ms 30 ms 10 5 ms
22800	11:44 697	11:21 344	11:51 223	10:53 173	10:25 138	2:14 113	1:52 97	1:27 86
24000	11:44 712	11:22 352	11:52 240	10:54 185	10:26 143	2:15 122	1:53 103	1:29 93
25200	11:45 741	11:23 375	11:53 252	10:55 193	10:27 146 148 ms 70 ms 13 2 ms	2:16 126	1:54 107	1:30 96
26400	11:46 786	11:24 395	11:54 264	10:56 203	10:30 160	2:17 131	1:55 113	1:31 100
27600	11:47 819	11:27 410	11:55 276	10:57 213	10:31 167	2:18 140	1:56 119	1:32 104
28800	11:48 856	11:28 429	11:56 287	11:30 217	10:32 171	2:19 145	1:57 124	1:33 110
30000	11:49 896	11:30 447	11:57 299	11:33 224	10:33 180	2:20 151	1:58 129	1:34 112
	2/7/73 (a.m.)		2/6/73 (a.m.)			2/5/73 (p.m.)		

2/5/73
5 min. test
@ 2:20 pm
no gaps
4669 buffers sent

2/6/73
8 min. test
at 11:58 am
no gaps
7141 buffers sent

2/7/73
18 min. test
@ 11:53 am
no gaps
16071 buffers sent

Results of a test for transmitting data from a continuous external process at UCSB (SEL 810-B computer) through the UCSB Host computer, over the ARPA network, and into a UCLA (site 65) Host discard socket (socket 9). Each test ≈ 30 sec.



from a real-time process operating as high as 30,000 bits/second. Further encouragement is given by three additional tests which were carried out at 30,000 bps and a buffer size of 2,016 bits. On 2/5/73 at 2:20 P.M. a 5-minute test was executed with no gaps. On 2/6/73 at 11:58 A.M. the same test was executed for 8 minutes with no gaps. The third test was conducted for 18 minutes on 2/7/73 at 11:53 A.M. with no gaps in the process.

The tests were not carried out often enough or over a long enough period of time to obtain any statistical results or predictions. The measurement task is made somewhat difficult by the fact that the state of the overall communications link is never repeatable from one test to the next. For example, it was found that a test which failed could usually be repeated successfully, even when it was carried out within 15 seconds of the previous test.

VII. CONCLUSIONS

Based upon the results of this experiment the following conclusions can be drawn:

1) High bit-rate real-time processes can use the ARPA Network to transmit data for relatively long periods of time.
2) Real-time processes accessing the Network through large-scale timesharing host computers can expect arbitrary delays or gaps, probably attributable to the host computers and not the Network.

3) Techniques for handling gaps of 1/2 to 1 second may be possible but 8 second gaps, as measured in this experiment, will cause extreme hardship on any real-time process.

This experiment has pointed up the need to conduct additional tests using a complete transmission link with actual data and with monitoring equipment at both the sending and receiving ends. Our current and future efforts are directed toward carrying out such experiments.

REFERENCES

[1] "Interface Message Processors for the ARPA Computer Network", Quarterly Technical Report No. 16, 1 Oct 1972 to 31 Dec 1972, Bolt, Beranek and Newman, Inc.

[2] Semiannual Technical Summary on Graphics, Lincoln Laboratory, Massachusetts Institute of Technology, Nov 1971.

[3] D.L. Retz, "An Interactive System for Signal Analysis: Design, Implementation, and Applications", CSL Report No 25, Computer Systems Lab, University of California, Santa Barbara, CA, 1972.